ZE**R**O TO HE**R**O
THE DEFINITIVE INTRO SERIES

What is R?

R is an open source language and environment for statistical computing and graphics. In addition to being free (as in speech, eg. open source) it is also free (as in free beer, yay!), which means you can download it and use it at home for free. It works on all major OS's including windows, mac, and linux.

Downloading and Installation

http://cran.r-project.org/ ← Here you can find the software available for download as well as links to introductions and tutorials.

R-Studio

At it's core, R is just a command line interface – just a lonely blinking cursor waiting for input. R-studio provides a fancier interface with some handy tools to make your life a lot easier. Like R, it is also free and open source software.

http://rstudio.org/download/desktop ← Here you can find r-studio for download. Make sure you've installed R first.

Fire up R-Studio and have a look:
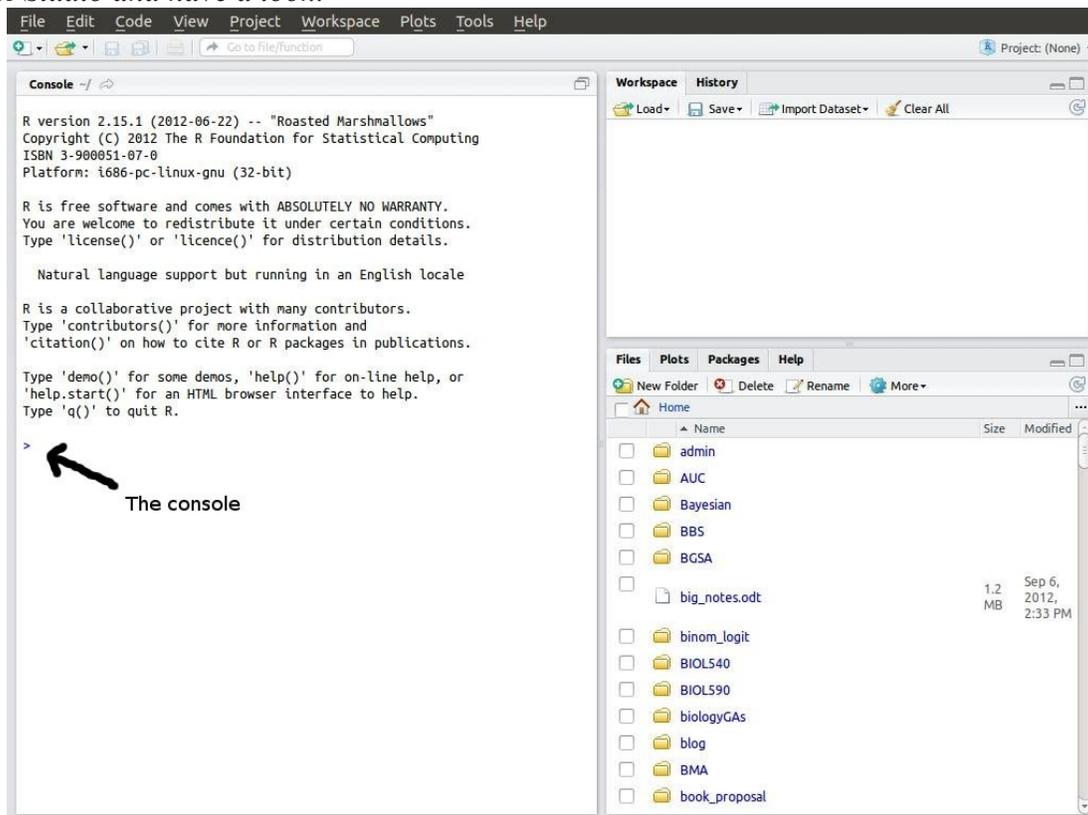
Pre-Workshop

The Console
You can type commands directly into the console. Try starting with some simple math.

```
> 2+2
[1] 4
```

Notice that the output is printed out right in the console. Try something a little more funky:

```
> cos(3.14159)
[1] -1
```

ProTip: in the console, use the ↑ & ↓ to cycle through previous commands (don't retype any more than you have to!)

You can also assign the output of a command to a variable (we'll call variables in R *objects*).

```
> a<-7*5
```

the `<-` operator is used to assign values to objects. Notice how it looks like an arrow pointing from the command `7*5` toward `a`. You can see the value of your new object a by typing it in to the console.

```
> a
[1] 35
```

Scripts
Typing directly into the console is cool and everything, but when we start getting into more complicated multi-part analyzes, we're going to want to have some way to keep track of what we've done. If we make an error at some point, we don't want to have to type everything in all over again. The solution is to use a *script*. A script is just a plain text file containing a list of commands. **Always write your commands into a script file.**

Open a new script using *File > New R Script* in Rstudio. To run a line from your script in the console, use C*trl+Enter*. Anything following a `#` on a line will be ignored by the console. Use `#` to insert comments about what your script is doing and to write notes to yourself about the analysis.

Functions
We have already seen a function when we ran the command `sin(3.14159)`. A function takes *arguments* in the parentheses, and returns a result. To get information about how to use different functions type `?<functionname>` (ex: `?sin`). Here are just a few of R's *many* built-in functions:

sum	unique	min	paste
mean	sqrt	max	head
read.csv	log	sd	length
cbind	exp	summary	str
rbind	rep	plot	names

The help pages that appear in the bottom right when you use ? Are themselves a little difficult to understand at first, but they all have the same structure so once you've seen a few they will start to make sense. Most have an *Examples* section at the bottom with sample code which you can run to get a feel for them.